Université de Montréal
École des Hautes Études Commerciales
École Polytechnique

Université de Montréal
C.P. 6128, Succursale Centre-ville,
Montréal (Québec) Canada H3C 3J7

Téléphone +1-514-343-7575
Télécopieur (Fax) +1-514-343-7121
crt@crt.umontreal.ca
www.crt.umontreal.ca/CRT/

CRT-2002-08

Centre de recherche sur les transports (C.R.T.)
Centre for Research on Transportation

THE GENERALIZED MINIMUM SPANNING TREE PROBLEM: POLYHEDRAL ANALYSIS AND BRANCH-AND-CUT ALGORITHM

by

Corinne Feremans
Martine Labbé
Gilbert Laporte

April 2002

PUBLICATION
CRT-2002-08

THE GENERALIZED MINIMUM SPANNING TREE PROBLEM: POLYHEDRAL ANALYSIS AND BRANCH-AND-CUT ALGORITHM*

by

Corinne Feremans[1,2]
Martine Labbé[1]
Gilbert Laporte[3]

* This research was supported in part by the Fonds pour la Formation à la Recherche dans l'Industrie et dans l'Agriculture, by the Région Bruxelles-Capitale, by the Direction générale des Relations Extérieures de la Région wallone de Belgique and by the Natural Sciences and Engineering Research Council of Canada (NSERC).

[1] Institut de Statistique et de Recherche Opérationnelle, Service d'Optimisation, C.P. 210/01, Université Libre de Bruxelles, boulevard du Triomphe, B-1050 Bruxelles, Belgium (mlabbe@smg.ulb.ac.be)

[2] Universiteit Maastricht, Faculty of Economics and Business Administration Department, Quantitative Economics, P.O. Box 616, 6200 MD Maastricht, The Netherlands (C.Feremans@KE.unimaas.nl)

[3] Canada Research Chair in Distribution Management, École des Hautes Études Commerciales, 3000 chemin de la Côte-Sainte-Catherine, Montréal H3T 2A7 Canada and Centre de recherche sur les transports, Université de Montréal, C.P. 6128, succursale Centre-Ville, Montréal H3C 3J7 Canada (gilbert@crt.umontreal.ca)

Dépôt légal – Bibliothèque nationale du Québec
Bibliothèque nationale du Canada, 2002

103-03055/

Abstract

This article presents a branch-and-cut algorithm for the Generalized Minimum Spanning Tree Problem (GMSTP). Given an undirected graph whose vertex set is partitioned into clusters, the GMSTP consists of determining a minimum cost tree including exactly one vertex per cluster. Applications of the GMSTP are encountered in telecommunications. An integer linear programming formulation is presented and new classes of valid inequalities are developed, several of which are proved to be facet defining. A branch-and-cut algorithm and a tabu search heuristic are developed. Extensive computational experiments show that instances involving up to 160 or 200 vertices can be solved to optimality, depending on whether edge costs are Euclidean or random.

Key words: generalized minimum spanning tree problem, network design, telecommunications, polyhedral analysis, branch-and-cut algorithm, tabu search heuristic.

Résumé

Cet article décrit un algorithme de séparation et coupes pour le problème de l'arbre recouvrant minimum généralisé (PARMG). Étant donné un graphe non orienté dont l'ensemble de sommet est partitionné en sous-ensembles, le PARMG consiste à déterminer un arbre de coût minimal contenant exactement un sommet par sous-ensemble. On retrouve des applications du PARMG en télécommunications. On présente une formulation par programmation linéaire en nombres entiers ainsi que de nouvelles familles d'inégalités valides dont plusieurs définissent des facettes. Un algorithme de séparation et coupes ainsi qu'une heuristique de recherche avec tabous sont développés. Des résultats de calcul extensifs révèlent que des instances comportant jusqu'à 160 ou 200 sommets peuvent être résolus de façon optimale, selon que les coûts sont euclidiens ou aléatoires.

Mots-clefs : problème de l'arbre recouvrant minimum généralisé, conception de réseaux, télécommunications, analyse polyédrale, algorithme de séparation et coupes, heuristique de recherche avec tabous.

1 Introduction

The purpose of this article is to present a model, a polyhedral analysis, and a branch-and cut algorithm for the *Generalized Minimum Spanning Tree Problem* (GMSTP) defined as follows. Let $G = (V, E)$ be an undirected graph where V is the vertex set partitioned into clusters V_k, $k \in K$, and $E \subseteq \{\{i, j\} : i, j \in V\}$ is the edge set. Each edge $e = \{i, j\} \in E$ has a cost $c_e \in \mathbb{R}^+$. The GMSTP is the problem of determining a minimum cost tree including *exactly* one vertex from each cluster. This problem was introduced by Myung, Lee and Tcha [16]. Another version of the problem, not considered in this study, is the case where the tree must include *at least* one vertex from each cluster (Ihler, Reich and Widmayer [14], Dror, Haouari and Chaouachi [4], Feremans, Labbé and Laporte [6]).

Applications for the GMSTP are encountered in telecommunications where metropolitan and regional networks must be interconnected by a tree containing a gateway from each network. For this internetworking, a vertex has to be chosen in each local network as a hub and the hub vertices must be connected via transmission links such as optical fiber (see Myung, Lee and Tcha [16]).

Using a reduction from the *Node Cover Problem*, Myung, Lee and Tcha [16] show that the GMSTP is strongly $\mathcal{NP}$-hard. These authors also provide four integer linear programming formulations. A branch-and-bound method is developed and applied to the exact resolution of instances involving up to 100 vertices. Faigle, Kern, Pop and Still [5] construct an alternative mixed integer linear programming formulation for the GMSTP which is used to solve instances containing up to 75 vertices. Pop, Kern and Still [18] provide a polynomial approximation algorithm for the GMSTP. Its worst-case ratio is bounded by 2ρ if the cluster size is bounded by ρ. This algorithm is derived from the method described in Magnanti and Wolsey [15] for the *Vertex Weighted Steiner Tree Problem*.

The remainder of this article is organized as follows. In Section 2, we model the GMSTP by means of a tight integer linear programming formulation, and we also develop several families of valid inequalities. A polyhedral analysis is conducted in Section 3. It is shown that several families of valid inequalities are facet defining. A branch-and-cut algorithm, a tabu search heuristic, and separation procedures are described in Section 4. Computational results, presented in Section 5, show that instances defined on complete graphs involving up to 200 vertices can be solved to optimality using the proposed methodology. Conclusions follow in Section 6.

2 Model and valid inequalities

Several integer linear programming formulations are available for the GMSTP. In a previous article (Feremans, Labbé and Laporte [7]), eight of these are analyzed and compared. It is shown that the following *Undirected Cluster Subpacking Formulation* is not only among the tightest in terms of the linear programming relaxation, but is also the most compact in the number of variables.

2.1 Undirected Cluster Subpacking Formulation

First define the binary variables.

$$x_e = x_{ij} = \begin{cases} 1 & \text{if } e = \{i,j\} \in E \text{ is selected in a solution,} \\ 0 & \text{otherwise,} \end{cases}$$

$$\text{and} \quad y_i = \begin{cases} 1 & \text{if } i \in V \text{ is selected in a solution,} \\ 0 & \text{otherwise.} \end{cases}$$

We use the vector notation $x = (x_{ij})$, $y = (y_i)$ and the notations $x(E') = \sum_{\{i,j\} \in E'} x_{ij}$ for $E' \subseteq E$ and $y(V') = \sum_{i \in V'} y_i$ for $V' \subseteq V$. The *incidence vector* of a subgraph $H = (V(H), E(H))$ of G is defined as (x, y), with binary components $x_e = 1$ if and only if $e \in E(H)$, and $y_i = 1$ if and only if $i \in V(H)$. We denote by $\delta(S) = \{e = \{i,j\} \in E : i \in S, j \in V \backslash S\}$, the subset of edges that cross S and by $E(S) = \{e = \{i,j\} \in E : i,j \in S\}$, the subset of edges with two end vertices in S. For $S, T \subseteq V$, define $E(S:T) = \{e = \{i,j\} \in E : i \in S, j \in T\}$. For simplicity, we will write $\delta(i)$ instead of $\delta(\{i\})$. We denote by $V_{h(i)}$ the cluster containing $i \in V$. We define $W = \{v \in V : |V_{h(v)}| = 1\}$, the set of vertices for which the cluster is a singleton. For $S \subseteq V$ we define $\mu(S) = |\{k : V_k \subseteq S\}|$, the number of clusters included in S, and $\eta(S) = |\{k : V_k \cap S \neq \emptyset\}|$, the number of clusters intersecting S. The formulation is then:

GMST Minimize $\sum_{e \in E} c_e x_e$

subject to

$$y(V_k) = 1 \qquad (k \in K), \tag{1}$$

$$x(E) = |K| - 1, \tag{2}$$

$$x(E(S)) \leq y(S) - 1 \quad (S \subset V, 2 \leq |S| \leq n-1, \mu(S) \neq 0), \tag{3}$$

$$x_e \in \{0,1\} \qquad e \in E, \tag{4}$$

$$y_i \in \{0,1\} \qquad i \in V. \tag{5}$$

In this formulation, constraints (1) and (2) ensure that the tree spans each cluster exactly once, while constraints (3) are Generalized Subtour Elimination constraints (GSEC). In what follows (3, k, S) refers to constraint (3) relative to cluster V_k and subset S. Note that two special cases of the GSEC are

$$x(E(\{i\} : V_k)) \leq y_i \qquad i \in V \backslash (W \cup V_k), \tag{6}$$

$$x(\delta(i)) \geq y_i \qquad i \in V. \tag{7}$$

Constraints (6) are simply a rewriting of (3) for $S = \{i\} \cup V_k$. Constraints (7) are also a rewriting of (3) for $S = V \backslash \{i\}$ since $|K| - 1 = x(E) = x(\delta(i)) + x(E(V \backslash \{i\}))$. Constraints (7) and (6) are used in Step 4 of the branch-and-cut algorithm.

2.2 Valid inequalities

We now present four families of valid inequalities for **GMST**.

Proposition 1 (Hammock inequalities). *Let $r, s \in V$ be such that $h(r) \neq h(s)$, $|V_{h(r)}| \geq 2, |V_{h(s)}| \geq 2$. Given an index set $\tilde{K} \subseteq K \backslash \{h(r), h(s)\}$ with $|\tilde{K}| \geq 2$, the inequality*

$$(|\tilde{K}| - 1)x_{rs} + y_r + y_s + x(\bigcup_{k \in \tilde{K}} E(V_k : (V_{h(r)} \cup V_{h(s)}) \backslash \{r,s\})) + x(E(V_{h(r)} \backslash \{r\} : V_{h(s)} \backslash \{s\})) \leq |\tilde{K}| + 1 \tag{8}$$

is valid for **GMST** *(see Figure 1).*

Proof. If $y_r = y_s = 1$, the constraints $x_{ij} \leq y_i, x_{ij} \leq y_j$ imply $x(\bigcup_{k \in \tilde{K}} E(V_k : (V_{h(r)} \cup V_{h(s)}) \backslash \{r,s\})) + x(E(V_{h(r)} \backslash \{r\} : V_{h(s)} \backslash \{s\})) = 0$. In this case (8) becomes $x_{rs} \leq 1$, which is valid for **GMST**. Otherwise $x_{rs} = 0$, and constraints (1) and $(3, \ell, \bigcup_{k \in \tilde{K}} V_k \cup V_{h(r)} \backslash \{r\} \cup V_{h(s)} \backslash \{s\}), \ell \in \tilde{K}$ imply (8). □

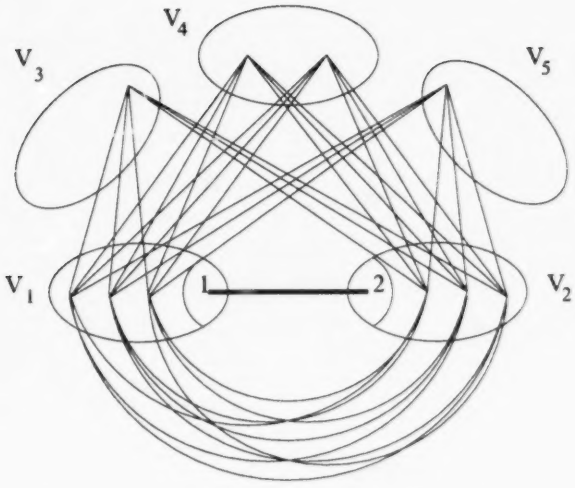

Figure 1: Support graph of the Hammock constraint for $r = 1, s = 2, h(r) = 1, h(s) = 2$ and $\tilde{K} = \{3, 4, 5\}$

Proposition 2 (Odd Cycle inequalities). *Consider $V_1, \ldots, V_m$, where m is odd, and define partitions $\{V_k^1, V_k^2\}$ for $k = 1, \ldots, m$. Then the inequality*

$$\sum_{k=1}^{m} x(E(V_k^1 : V_{(k \bmod m)+1}^2)) \leq \lfloor m/2 \rfloor \tag{9}$$

is valid for **GMST** *(see Figure 2).*

Proof. The validity of (9) follows from the fact that a feasible GMSTP solution contains at most one vertex per cluster. □

Proposition 3 (Odd Clique-Matching inequalities). *Let $V_1, \ldots, V_m$ be m clusters where m is odd, and define partitions $\{V_k^\ell\}$ for $k = 1, \ldots, m$ and $\ell = 1, \ldots, m-1$. Then*

$$\sum_{k=1}^{m-1} \sum_{\ell=k}^{m-1} x(E(V_k^\ell : V_{\ell+1}^k)) \leq \lfloor m/2 \rfloor \tag{10}$$

is a valid inequality for **GMST**.

Proof. The validity of (10) follows from the fact that a feasible GMSTP solution contains at most one vertex per cluster. □

The following classes of valid inequalities can be derived from a *Stable Set* relaxation of the GMSTP. First observe that any two edges incident to different vertices of the same cluster cannot belong to a feasible GMSTP solution. Such edges are said to be *in conflict*. It is useful to define a *conflict graph* $G_C = (V_C, E_C)$, where each edge of E is a vertex of G_C, and an edge of E_C is defined between e_1 and e_2 of E if and only if e_1 and e_2 are in conflict. Any GMSTP solution is a *stable set* in G_C, i.e., a set of elements of V_C not linked by any edge of E_C. Thus any valid inequality for the *Stable Set Problem* defined

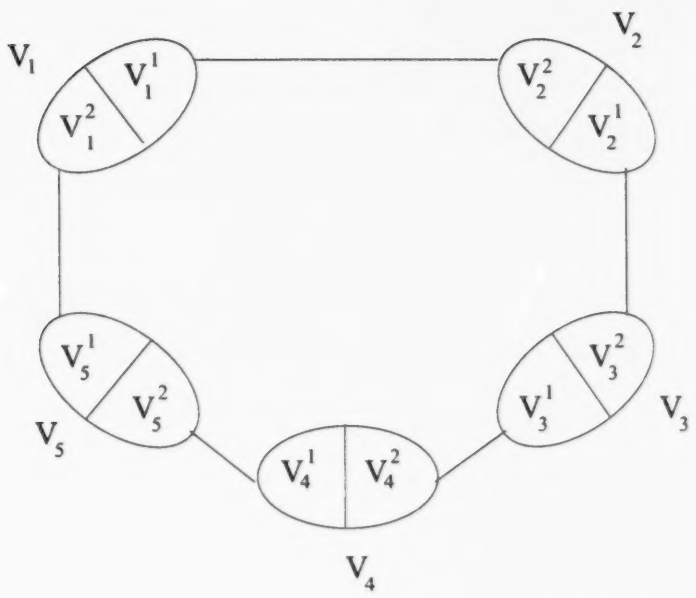

Figure 2: Support graph of the Odd Cycle inequality for $m = 5$

on G_C is valid for the GMSTP defined on G. Several such inequalities are described in Padberg [17]. We found the following *Odd Hole* inequaliities particularly effective in our branch-and-cut algorithm.

Proposition 4 (Odd Hole inequalities). *Let $F \subseteq E$ ($|F|$ odd, $|F| \geq 3$) inducing a cordless cycle in G_C. Then the inequality*

$$x(F) \leq \lfloor |F|/2 \rfloor \tag{11}$$

is valid for **GMST**.

Proof. Follows directly from the definition of G_C. □

3 Polyhedral analysis

In order to investigate polyhedral properties of the above constraints, first define the polytope $\mathcal{P}(G)$ associated with **GMST** on a graph G: $\mathcal{P}(G) = \text{conv}\,\{(x, y) \in \{0,1\}^{|E|+|V|}$, such that (1), (2) and (3) hold$\}$. Let v_k be a vertex arbitrarily chosen in each V_k for $k \in K$, we denote by $star(i, L)$, $i \in V, L \subseteq K$, the subset of edges of G in $\delta(i) \cap \delta(\{v_k : k \in L \backslash \{h(i)\}\})$. Our analysis holds for the case where $|K| \geq 5$ and G is a complete $|K|$-partite graph.

3.1 Preliminary results

We first determine the dimension of $\mathcal{P}(G)$ and we introduce two lemmas used in the following subsections. Definition 1 is taken from Fischetti, Salazar and Toth [8].

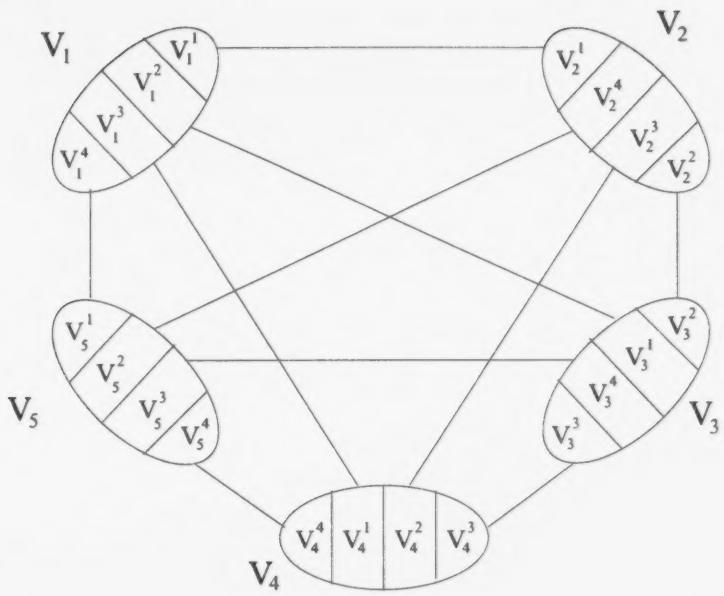

Figure 3: Support graph of the Odd Clique-Matching inequality for $m = 5$

Definition 1. *Given a valid inequality $\alpha x \leq \beta y + \gamma$ for $\mathcal{P}(G)$, let $\mathcal{H}(\alpha, \beta, \gamma) = \mathcal{P}(G) \cap \{(x, y) \in \mathbb{R}^{|E|+|V|} : \alpha x = \beta y + \gamma\}$ denote the face of $\mathcal{P}(G)$ induced by $\alpha x \leq \beta y + \gamma$. Let $v \in V \backslash W$ be an arbitrary but fixed vertex. The* v-restriction *of $\alpha x \leq \beta y + \gamma$ is the inequality obtained from $\alpha x \leq \beta y + \gamma$ by dropping the variables y_v and x_e for all $e \in \delta(v)$. Similarly, let $i, j \in V \backslash W$ be arbitrary but fixed vertices. The* ij-restriction *of $\alpha x \leq \beta y + \gamma$ is the inequality obtained from $\alpha x \leq \beta y + \gamma$ by dropping the variables y_i, y_j and x_e for all $e \in \delta(i) \cup \delta(j)$.*

Lemma 1. *For every valid inequality $\alpha x \leq \beta y + \gamma$ for $\mathcal{P}(G)$ and every vertex $v \in V \backslash W$, the dimension of $\mathcal{H}(\alpha, \beta, \gamma)$ is greater than or equal to the dimension of the face of $\mathcal{P}(G \backslash \{v\})$ induced by its v-restriction, plus the rank of the matrix formed by the coordinates (x, y), restricted to x_e for $e \in \delta(v)$ and y_v, of the extreme points of $\mathcal{H}(\alpha, \beta, \gamma)$ with $y_v = 1$.*

Proof. Let X be the matrix in which every row is an extreme point of $\mathcal{H}(\alpha, \beta, \gamma)$. Since $\mathcal{H}(\alpha, \beta, \gamma)$ is contained in a hyperplane not passing through the origin (e.g. induced by constraint (2)), the dimension of $\mathcal{H}(\alpha, \beta, \gamma)$ coincides with the rank of X, minus 1. Since $v \notin W$, X can be partitioned into $X = \begin{pmatrix} X_{11} & 0 & 0 \\ X_{21} & X_{22} & 1 \end{pmatrix}$, where the last column corresponds to variable y_v, and the columns of X_{22} correspond to variables $x_e \in \delta(v)$. Since rank $X \geq$ rank X_{11} + rank $(X_{22} \quad 1)$, and the dimension of the face of $\mathcal{P}(G \backslash \{v\})$ induced by the v-restriction of $\alpha x \leq \beta y + \gamma$ is equal to the rank of X_{11}, minus 1, then $\dim(\mathcal{H}(\alpha, \beta, \gamma)) \geq$ dim (face of $\mathcal{P}(G \backslash \{v\})$ induced by the v-restriction) + rank $(X_{22} \quad 1)$. □

Proposition 5. *The dimension of $\mathcal{P}(G)$ is equal to $|E| + |V| - |K| - 1$.*

Proof. Clearly, $\dim(\mathcal{P}(G)) \leq |E| + |V| - |K| - 1$ since the equations $y(V_k) = 1, k \in K$ and $x(E) = |K| - 1$ are valid for $\mathcal{P}(G)$ and are linearly independent. We use induction on $\rho = |V| - |K|$ to prove the reverse inequality. When $\rho = 0$, then all clusters are singletons and the claim is true (see Goemans [10]). Assume now that the claim holds for $\rho = \bar{\rho}$, and consider any GMSTP polytope with $\rho = \bar{\rho} + 1$. Then, there exists a vertex $v \in V \backslash W$. This induction hypothesis with Lemma 1 for the trivial valid inequality $0 \leq 0$ leads to $\dim(\mathcal{P}(G)) \geq |E \backslash \delta(v)| + |V \backslash \{v\}| - |K| - 1 + \text{rank } (X_{22} \quad 1)$. Thus, if we show that rank $(X_{22} \quad 1) = |\delta(v)| + 1$, we are done. This is equivalent to exhibiting $|\delta(v)| + 1$ incidence vectors of a GMSTP feasible solution containing vertex v such that their restriction to x_e for all $e \in \delta(v)$ and y_v are linearly independent. Let v_k be a vertex arbitrarily chosen in V_k for each $k \in K$. For every vertex $p \in V \backslash V_{h(v)}$, we consider $star(p, K)$ where $v_{h(v)} = v$. These trees show that rank $(X_{22}) \geq \text{rank}(I_{|\delta(v)| \times |\delta(v)|}) = |\delta(v)|$. Since $|K| \geq 5$, there exist $p, q \in V$ such that $h(p) \neq h(q) \neq h(v)$. The restriction to x_e for all $e \in \delta(v)$ and y_v of the incidence vector of the GMSTP feasible solution $\{\{v, p\}\} \cup star(q, K \backslash \{h(p)\})$ is obviously linearly independent from the previous $|\delta(v)|$ vectors.

□

Lemma 2 (Vertex cloning for $\mathcal{P}(G)$). *Let $\alpha x \leq \beta y + \gamma$ be a valid inequality inducing a proper face of $\mathcal{P}(G)$. If there exists a vertex $v \in V \backslash W$ such that the two following conditions hold:*

1. *the v-restriction of this inequality is facet defining for $\mathcal{P}(G \backslash \{v\})$, and*
2. *there exist $|\delta(v)|+1$ incidence vectors of a GMSTP solution of the graph G containing vertex v, satisfying $\alpha x = \beta y + \gamma$, and such that their restriction to x_e for all $e \in \delta(v)$ and y_v are linearly independent,*

then $\alpha x \leq \beta y + \gamma$ induces a facet of $\mathcal{P}(G)$.

Proof. Since $\mathcal{H}(\alpha, \beta, \gamma)$ is a proper face of $\mathcal{P}(G)$, then $\dim(\mathcal{H}(\alpha, \beta, \gamma)) \leq |E| + |V| - |K| - 2$. Hypothesis 1 and Proposition 5 imply that $\dim(v\text{-restriction}) = |E \backslash \delta(v)| + |V \backslash v| - |K| - 2 = |E| + |V| - |K| - |\delta(v)| - 3$. Using the notation of Lemma 1, hypothesis 2 is equivalent to rank $(X_{22} \quad 1) \geq |\delta(v)| + 1$. Lemma 1 implies $\dim(\mathcal{H}(\alpha, \beta, \gamma)) \geq |E| + |V| - |K| - 2$. Therefore $\dim(\mathcal{H}(\alpha, \beta, \gamma)) = |E| + |V| - |K| - 2$ and $\alpha x \leq \beta y + \gamma$ defines a facet of $\mathcal{P}(G)$. □

Another natural corollary to Lemma 1 is the following.

Lemma 3 (Vertex cloning for $\mathcal{P}(G)$). *Let $\alpha x \leq \beta y + \gamma$ be a valid inequality inducing a proper face of $\mathcal{P}(G)$. If there exists a vertex $v \in V \backslash W$ such that the two following conditions hold:*

1. *the v-restriction of this inequality is trivial (namely, the face induced by the v-restriction leads to $\mathcal{P}(G)$) and,*
2. *there exist $|\delta(v)|$ incidence vectors of a GMSTP solution of the graph G containing vertex v, satisfying $\alpha x = \beta y + \gamma$, and such that their restriction to x_e for all $e \in \delta(v)$ and y_v are linearly independent,*

then $\alpha x \leq \beta y + \gamma$ *induces a facet of* $\mathcal{P}(G)$.

Proof. Similar to the proof of Lemma 2. □

We are now ready to study several families of facet-inducing valid inequalities for $\mathcal{P}(G)$. We first study under which conditions trivial inequalities are facet defining.

3.2 Bound inequalities and Generalized Subtour Elimination constraints

We first prove that Bound inequalities and GSEC are facet defining.

Proposition 6 (Bound inequalities).

1. *The inequality* $x_e \geq 0$, $e \in E$, *defines a facet of* $\mathcal{P}(G)$.
2. *The inequality* $y_i \geq 0$, $i \in V$ *does not define a facet of* $\mathcal{P}(G)$.
3. *The inequality* $y(V_k) \leq 1, k \in K$ *does not define a facet of* $\mathcal{P}(G)$.

Proof.

1. Let $e = \{i, j\}$. Let v_k be a vertex arbitrarily chosen in each V_k for $k \in K$. We first prove by induction on $\rho = |V| - |K|$ that $x_e \geq 0$ induces a facet of $\mathcal{P}(G)$. When $\rho = 0$, all clusters are singletons and the claim is true (Goemans [10]). Assume now that the claim holds for $\rho = \bar{\rho}$ and consider any GMSTP instance with $\rho = \bar{\rho} + 1$. There exists a vertex $v \notin \{i, j\} \cup W$, assume that $v_{h(v)} = v$. By Lemma 2 and the induction hypothesis, if we exhibit $|\delta(v)| + 1$ GMSTP feasible solutions containing vertex v and satisfying the assumptions of Lemma 2, we are done.
   Two cases can occur.
   1.1. If $v \in V_{h(i)} \cup V_{h(j)}$, note that $y_v = 1$ implies that $x_e = 0$. Assume that $v \in V_{h(i)}$. In this case, the $|\delta(v)|$ vectors correspond to the GMSTP feasible solutions $star(p, K)$ for $p \in V \backslash V_{h(i)}$. Since $|K| \geq 3$ there exist a vertex p such that $h(p) \neq h(i), h(j)$, and the last needed vector can be $star(p, K \backslash \{h(j)\}) \cup \{\{v, v_j\}\}$. Obviously, the restricted vectors are linearly independent.
   1.2. If $v \notin V_{h(i)} \cup V_{h(j)}$, for every $p \in V \backslash (V_{h(v)} \cup V_{h(i)} \cup V_{h(j)})$, the GMSTP feasible solution $star(p, K)$ where $v_{h(i)} = i, v_{h(j)} = j$ does not contain the edge $e = \{i, j\}$. Since we have assumed that $|K| \geq 5$, for $p \in V_{h(i)} \cup V_{h(j)}$, there exists a vertex $q \in V \backslash (V_{h(v)} \cup V_{h(i)} \cup V_{h(j)})$ such that the GMSTP feasible solution $star(p, K \backslash \{h(i), h(j)\}) \cup \{\{q, v_k\}\}$ where $k \in \{h(i), h(j)\} \backslash \{h(p)\}$, does not contain the edge $e = \{i, j\}$. Their incidence vectors are linearly independent when their coordinates are restricted to $x_e, e \in \delta(v)$ and y_v. The last incidence vector of the $star(i, K \backslash \{h(j)\}) \cup \{\{v, j\}\}$ completes the requested set.
2. The inequality $y_i \geq 0$ does not define a facet of $\mathcal{P}(G)$. Indeed, $\mathcal{P}(G) \cap \{(x, y) : y_i = 0\} \subset \mathcal{P}(G) \cap \{(x, y) : x(E(\{i\} : V_k)) = 0\} \subset \mathcal{P}(G)$ where $k \in K, k \neq h(i)$.

3. Constraints (1) do not define facets of $\mathcal{P}(G)$ since they are tight for all points of $\mathcal{P}(G)$.

□

We now revert our attention to the cycle elimination constraints $(3, k, S), k \in K, V_k \subset S \subset V$.

Proposition 7 (GSEC for $\mathcal{P}(G)$). *Let $k \in K$, $V_k \subset S \subset V, |S| \geq 2$. Then inequality $(3, k, S)$ is facet defining for $\mathcal{P}(G)$ if and only if any of the following four conditions holds:*

1. $\eta(S) \geq 3$ *and* $\eta(V \backslash S) \geq 2$,
2. $S \subseteq W, |S| = 2$,
3. $S = \{v\} \cup V_k$ *with* $v \notin W \cup V_k$,
4. $S = V \backslash \{v\}, v \in V$.

Proof. Let v_k be a vertex arbitrarily chosen in each V_k for $k = 1, \ldots, |K|$. Assume that $v_{h(v)} = v$.

1. The *sufficient* condition is proven in four steps.

   1.1. We first show that if condition 1 is satisfied, then constraint $(3, k, S)$ is facet defining for $\mathcal{P}(G)$. Since $x(E(S)) \leq |S| - 1$ defines a facet of $\mathcal{P}(G)$ when $V = W$ and G is complete (see Goemans [10]), using Lemma 2, it is sufficient to show that for any given $v \notin W$, rank(X_{22} 1) $\geq |\delta(v)| + 1$. To this end, we exhibit $|\delta(v)| + 1$ GMSTP feasible solutions containing vertex v such that their restriction to S remains a tree and the restriction of their incidence vectors to y_v and x_e for all $e \in \delta(v)$ are linearly indepe dent. There are two cases:

      1.1.1. Vertex $v \notin S \cup W$. Since $\eta(S) \geq 3$, for each $p \notin S \cup V_{h(v)}$, there exists $q \in S \backslash (V_{h(v)} \cup V_{h(p)})$ such that $V_{h(q)} \subset S$. For each $p \notin S \cup V_{h(v)}$, construct the following GMSTP feasible solution $\{\{v, p\}\} \cup star(q, K \backslash \{h(v)\})$ where $v_p = p$. For each $p \in S \backslash V_{h(v)}$, construct the following GMSTP feasible solution $star(p, K)$. We need one more vector . Since $\eta(V \backslash S) \geq 2$ and $\eta(S) \geq 3$, there exists a vertex $r \in V \backslash (S \cup V_{h(v)} \cup V_{h(q)})$. One GMSTP feasible solution that fits our requirement is $star(q, K \backslash \{h(r)\}) \cup \{\{v, r\}\}$. This implies that for $v \notin S \cup W$, the conditions of Lemma 2 are satisfied.

      1.1.2. Vertex $v \in S \backslash W$. For each $p \notin S \cup V_{h(v)}$, construct the GMSTP feasible solution $\{\{v, p\}\} \cup star(q, K \backslash \{h(p)\})$. For each $p \in S \backslash V_{h(v)}$, construct the following GMSTP feasible solution $star(p, K)$. We need one more vector. Since $\eta(S) \geq 3$, there exists $p \in S \backslash V_{h(v)}$ such that $h(p) \neq h(q) \neq h(v)$. One GMSTP feasible solution that fits our requirement is $\{\{v, p\}\} \cup star(q, K \backslash \{h(p)\})$. This implies that the constraints $x(E(S)) \leq y(S) - 1$ with $\mu(S) \neq 0$ are facet defining for $\mathcal{P}(G)$ when $\eta(S) \geq 3$ and $\eta(V \backslash S) \geq 2$.

1.2. Assume that condition 2 is satisfied. Since $x_e \leq 1$ is facet defining for $\mathcal{P}(G)$ when $V = W$ (see Goemans [10]), it is sufficient to exhibit $|\delta(v)| + 1$ GMSTP feasible solutions satisfying the conditions of Lemma 2. Let $e = \{r, s\}$, $v_{h(r)} = r$, $v_{h(s)} = s$, $v \notin W$ $(h(v) \neq h(r) \neq h(s))$. The $|\delta(v)|$ first vectors correspond to the GMSTP feasible solution $star(p, K)$ if $p \in \{r, s\}$ and $\{\{r, s\}\} \cup star(p, K \backslash \{h(s)\})$ otherwise. Since $|K| \geq 5$, there exists $p \in V$ such that $h(p) \neq h(v), h(r), h(s)$, and the last required vector can be, $\{\{r, s\}, \{r, v\}\} \cup star(p, K \backslash \{h(r), h(s)\})$ and we are done.

1.3. Assume that condition 3 is satisfied. In this case, $(3, k, S)$ can be rewritten as $x(E(\{v\} : V_k)) \leq y_v$. The v-restriction of this constraint is the trivial inequality $0 \leq 0$, so according to Lemma 3 it is sufficient to exhibit $|\delta(v)|$ GMSTP feasible solutions containing vertex v and satisfying the assumptions of Lemma 3. The trees $star(p, K)$ for $p \in V_k$ together with the trees $star(v_k, K) \backslash \{\{v_k, p\}\}$ for $p \in V \backslash (V_{h(v)} \cup V_k)$ satisfy the requirements.

1.4. Assume that condition 4 is satisfied. In this case, constraint $(3, k, S)$ can be rewritten $y_v \leq x(\delta(v))$.

1.4.1. If $v \notin W$, we can apply Lemma 3. The v-restriction of this constraint is the trivial inequality $0 \leq 0$. It remains then to exhibit $|\delta(v)|$ GMSTP feasible solutions containing v and satisfying $x(\delta(v)) = 1$ and the conditions of Lemma 3. The trees $star(p, K)$ for $p \in V \backslash V_{h(v)}$ fulfill the requirements.

1.4.2. If $v \in W$, constraint $(3, k, S)$ can be rewritten $x(\delta(v)) \geq 1$. Since the inequality is facet defining for $\mathcal{P}(G)$ when $V = W$, using the same argumentation as before, for a fixed $u \in V \backslash W$ we need to exhibit $|\delta(u)| + 1$ vectors satisfying Lemma 2. The trees $star(p, K)$ for $p \in V \backslash (V_{h(u)} \cup \{v\})$ verify the conditions. Let $p, q \in V$ such that $h(p) \neq h(q) \neq h(u) \neq h(v)$, the trees $star(q, K \backslash \{h(v)\}) \cup \{\{u, v\}\}$ and $star(q, K \backslash \{h(p)\}) \cup \{\{u, p\}\}$ lead to the last needed vectors.

2. We now prove the *necessary* condition. If none of the four conditions 1, 2, 3, 4 is satisfied, then one of the following two cases must hold.

2.1. Either $V_k \subseteq S \subseteq V_k \cup V_\ell$ for $\ell \neq k$ and $|S \backslash V_k| \geq 2$. Let u be a vertex in $S \backslash V_k$. We observe that the vectors of $\mathcal{P}(G)$ on the face induced by constraint $(3, k, S)$ belong to the hyperplane defined by $x(E(\{u\} : V_k)) = y_u$.

2.2. Or $V \backslash S \subseteq V_l, l \neq k$ with $|V \backslash S| \geq 2$. Let u be a vertex in $V \backslash S$. In this case $(3, k, S)$ can be rewritten as $y(V \backslash S) \leq x(\delta(S))$. We then note that the vectors of $\mathcal{P}(G)$ on the face induced by constraint $(3, k, S)$ are included in the hyperplane defined by $x(\delta(u)) = y_u$.

□

3.3 Hammock inequalities

Hammock inequalities also induce facets of $\mathcal{P}(G)$. To prove this result, two technical lemmas are needed.

Lemma 4 (Two-vertex cloning for $\mathcal{P}(G)$). *Let $\alpha x \leq \beta y + \gamma$ be a valid inequality inducing a proper face of $\mathcal{P}(G)$. If there exist a pair of vertices $i, j \in V \backslash W$ such that all of the following three conditions hold:*

1. *if $h(i) = h(j)$ then $V_{h(i)} \backslash \{i, j\} \neq \emptyset$,*
2. *the ij-restriction of $\alpha x \leq \beta y + \gamma$ is facet defining for $\mathcal{P}(G \backslash \{i, j\})$,*
3. *there exist $|\delta(i) \cup \delta(j)| + 2$ GMSTP feasible solutions for the graph G containing at least one vertex from $\{i, j\}$ and whose incidence vectors are such that their restriction to x_e for all $e \in \delta(i) \cup \delta(j)$ and y_i, y_j are linearly independent,*

then $\alpha x \leq \beta y + \gamma$ induces a facet of $\mathcal{P}(G)$.

Proof. Since $\mathcal{H}(\alpha, \beta, \gamma)$ is a proper face of $\mathcal{P}(G)$ then $\dim(\mathcal{H}(\alpha, \beta, \gamma)) \leq |E| + |V| - |K| - 2$. Hypotheses 1, 2 and Proposition 5 imply that $\dim(ij\text{-restriction}) = |E \backslash (\delta(i) \cup \delta(j))| + |V \backslash \{i, j\}| - |K| - 2 = |E| + |V| - |K| - |\delta(i) \cup \delta(j)| - 4$. Let $(X_{22} \quad Y_i \quad Y_j)$ be such that the columns of X_{22} represent the coordinates $x_e \in \delta(i) \cup \delta(j)$ and Y_i, Y_j correspond to the coordinates y_i, y_j of the extreme points of $\mathcal{H}(\alpha, \beta, \gamma)$, respectively, with $y_i + y_j \geq 1$. Hypothesis 3 is equivalent to $\text{rank}(X_{22} \quad Y_i \quad Y_j) \geq |\delta(i) \cup \delta(j)| + 2$. Using the same argument as in Lemma 1, it implies $\dim(\mathcal{H}(\alpha, \beta, \gamma)) \geq |E| + |V| - |K| - 2$. Therefore $\dim(\mathcal{H}(\alpha, \beta, \gamma)) = |E| + |V| - |K| - 2$ and $\alpha x \leq \beta y + \gamma$ defines a facet of $\mathcal{P}(G)$. □

Lemma 5. *Let $V_{h(r)} = \{r, t\}, V_{h(s)} = \{s, u\}$, and $|V_k| = 1$ for $k \in \tilde{K} \subseteq K \backslash \{h(r), h(s)\}$, $|\tilde{K}| \geq 2$. Then, there exist $5 + 2|\tilde{K}|$ incidence vectors (x, y) of GMSTP feasible solutions satisfying constraint (8) relative to $\tilde{K}$ as an equality and such that their restrictions to variables x_e for $e \in (\delta(t) \cup \delta(u)) \cap E(\bigcup_{k \in \tilde{K}} V_k \cup V_{h(r)} \cup V_{h(s)})$ and y_i for $i \in \{t, u\}$ are linearly independent, and $x_e = 0$ for $e \in (\delta(t) \cup \delta(u)) \backslash E(\bigcup_{k \in \tilde{K}} V_k \cup V_{h(r)} \cup V_{h(s)})$.*

Proof. Let v_k be a vertex arbitrarily chosen in each V_k for $k = 1, \ldots, |K|$. The proof uses induction on $\rho = |\tilde{K}|$.

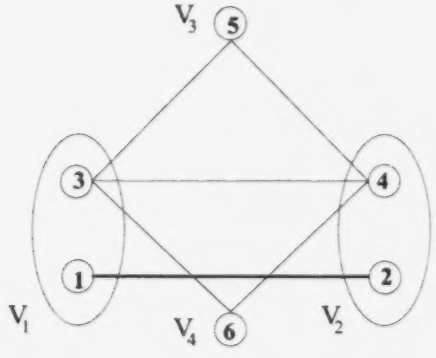

Figure 4: Support graph of the Hammock constraint for $r = 1, s = 2, h(r) = 1, h(s) = 2, V_1 = \{1, 3\}, V_2 = \{2, 4\}$, and $|V_k| = 1, k \in \tilde{K}, \tilde{K} = \{3, 4\}$

First, if $|\tilde{K}| = 2$, let $\bigcup_{k \in \tilde{K}} V_k = \{v_\ell, v_m\}$. The following nine GMSTP feasible solutions satisfy the requirements:

$$star(t, \tilde{K} \cup \{h(s)\}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})) \quad \text{where } v_{h(s)} = s, \tag{12}$$

$$star(t, \tilde{K} \cup \{h(s)\}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})) \quad \text{where } v_{h(s)} = u, \tag{13}$$

$$star(u, \tilde{K}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(s)\})) \quad \text{where } v_{h(r)} = t, \tag{14}$$

$$\{\{t, v_m\}\} \cup star(u, \tilde{K}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})), \tag{15}$$

$$star(u, \tilde{K} \cup \{h(r)\}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})) \quad \text{where } v_{h(r)} = r, \tag{16}$$

$$\{\{r, v_m\}\} \cup star(u, \tilde{K}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})), \tag{17}$$

$$\{\{t, v_\ell\}\} \cup star(u, \tilde{K} \backslash \{\ell\}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})), \tag{18}$$

$$star(u, \tilde{K} \cup \{h(r)\}) \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})) \quad \text{where } v_{h(r)} = t, \tag{19}$$

$$star(t, \tilde{K}) \cup \{\{s, v_m\}\} \cup star(v_\ell, K \backslash (\tilde{K} \cup \{h(r), h(s)\})). \tag{20}$$

Assume that the required vectors exist for all $\tilde{K}$ with $|\tilde{K}| \leq \rho$. Let $I \subseteq K \backslash \{h(r), h(s)\}$, we must show that the claim still holds for $\tilde{K} = I \cup \{\tilde{k}\}$ with $|I| = \rho$, $\tilde{k} \in K \backslash (I \cup \{h(r), h(s)\})$. We first transform the $5+2|I|$ vectors satisfying the conditions for I into vectors satisfying the conditions for $\tilde{K} = I \cup \{\tilde{k}\}$. If $y_t = 1$, then we set $x_e = 1$ for $e = \{t, v_{\tilde{k}}\}$. Otherwise, we set $x_e = 1$ for $e = \{u, v_{\tilde{k}}\}$. The induction hypothesis implies that the restrictions of these incidence vectors to x_e for $e \in (\delta(t) \cup \delta(u)) \cap E(\bigcup_{k \in I \cup \{\tilde{k}\}} V_k \cup V_1 \cup V_2)$ and y_i for $i \in \{t, u\}$ are linearly independent. To conclude, we exhibit two linearly independent incidence vectors of GMSTP feasible solutions such that their restriction to x_e for $e \in (\delta(t) \cup \delta(u)) \cap E(\bigcup_{k \in I \cup \{\tilde{k}\}} V_k \cup V_{h(r)} \cup V_{h(s)})$ and y_i for $i \in \{t, u\}$ are linearly independent from the previous $5 + 2|I|$ vectors and satisfy constraint (8) relative to $I \cup \{\tilde{k}\}$ to equality and $x_e = 0$ for $e \in (\delta(t) \cup \delta(u)) \backslash E(\bigcup_{k \in I \cup \{\tilde{k}\}} V_k \cup V_{h(r)} \cup V_{h(s)})$. Assume from now on that $v_{h(s)} = u$. The following two GMSTP feasible solutions satisfy the requirements:

$$star(t, I \cup \{h(s)\}) \cup star(v_{\tilde{k}}, K \backslash (I \cup \{h(r)\})), \tag{21}$$

$$star(t, I \cup \{\tilde{k}\}) \cup star(v_{\tilde{k}}, K \backslash (I \cup \{h(r)\})). \tag{22}$$

These trees clearly belong to the face induced by (8) relative to $\tilde{K} = I \cup \{\tilde{k}\}$. It remains to show that the restrictions of these two vectors form, together with the restrictions of the first $5 + 2|I|$ vectors, a set of linearly independent vectors. To conclude the proof, we subtract a particular linear combination of the previous $5+2|I|$ restrictions from (21) and (22). It then remains to show that the restriction to x_e for $e \in (\delta(t) \cup \delta(u)) \cap E(\bigcup_{k \in I \cup \{\tilde{k}\}} V_k \cup V_{h(r)} \cup V_{h(s)})$ and y_i for $i \in \{t, u\}$ of these two vectors have non-zero coordinates only for $e \in \{\{t, v_{\tilde{k}}\}, \{u, v_{\tilde{k}}\}\}$. For this purpose, consider the following three GMSTP feasible solutions:

$$star(t, I \cup \{h(s)\}) \cup star(v_{\tilde{k}}, K \backslash (I \cup \{h(u), h(s)\})), \tag{23}$$

$$star(u, I \cup \{h(r)\}) \cup star(v_{\tilde{k}}, K \backslash (I \cup \{h(s)\})), \tag{24}$$

$$\{\{t, v_{\tilde{k}}\}, \{t, v_i\}\} \cup star(u, I) \cup star(v_{\tilde{k}}, K \backslash (I \cup \{h(r), h(s)\})) \quad i \in I. \tag{25}$$

Denote by g_k, the incidence vector of (21) to (24) and by g_{25_i} the incidence vector of (25), $i \in I$. Then the combinations $g_{21} - g_{23}$ and $g_{22} + (|I|(g_{24} - g_{23}) - \sum_{i \in I} g_{25_i})/(|I| + 1)$ satisfy the requirements. Indeed, the restrictions of these two incidence vectors to x_e for $e \in (\delta(t) \cup \delta(u)) \cap E(\bigcup_{k \in I \cup \{\tilde{k}\}} V_k \cup V_{h(r)} \cup V_{h(s)})$ and y_i for $i \in \{t, u\}$ have non-zero coordinates only for $x_e, e \in \{(t, v_{\tilde{k}}), (u, v_{\tilde{k}})\}$. More precisely, the coordinates $x_e, e \in \{(t, v_{\tilde{k}}), (u, v_{\tilde{k}})\}$ for these two combinations are respectively $(-1, 1)$ and $(1/|\tilde{K}|, 1)$ which are linearly independent. □

Proposition 8. *The Hammock inequalities (8) are facet defining for $\mathcal{P}(G)$.*

Proof. The proof is divided into two parts. First, we will show that if $|V_{h(r)}| = |V_{h(s)}| = 2$ and $|V_k| = 1$ for $k \in \tilde{K}, |\tilde{K}| \geq 2$, then the claim holds. Let v_k be a vertex arbitrarily chosen in each V_k for $k = 1, \ldots, |K|$.

Using Lemma 4 and Proposition 7-2, we must exhibit $(|\delta(t) \cup \delta(u)| + 2)$ GMSTP feasible solutions containing at least one vertex from $\{t, u\}$ and satisfying (8) as an equality, and such that the restriction of their incidence vectors to y_t, y_u and $x \in \delta(t) \cup \delta(u)$ are linearly independent. For every vertex $v \in V \backslash (\bigcup_{k \in \tilde{K}} V_k \cup V_{h(r)} \cup V_{h(s)})$, one can construct two GMSTP feasible solutions $star(t, \tilde{K} \cup \{h(s)\}) \cup star(v, K \backslash (\tilde{K} \cup \{h(r), h(s)\}) \cup \{\{v, j\}\}$ for $j \in \{t, u\}$ where $v_{h(s)} = u$ satisfying these requirements. This yields $|\delta(t) \cup \delta(u)| - 2|\tilde{K}| - 3$ of the required vectors. Lemma 5 gives the last needed vectors and this concludes the first part of the proof.
In the second part, we use induction on $\rho = |(\bigcup_{k \in \tilde{K}} V_k \cup V_{h(r)} \cup V_{h(s)}) \backslash \{r, s\}| - |\tilde{K}| - 2$. The first part of the proof shows that the claim is true when $\rho = 0$. Assume now that the claim holds for $\rho \leq \bar{\rho}$ and consider any GMSTP polytope with $\rho = \bar{\rho} + 1$. Then there exists a vertex $v \in (\bigcup_{k \in \tilde{K}} V_k \cup V_{h(r)} \cup V_{h(s)}) \backslash (W \cup \{r, s\})$. This induction assumption together with Lemma 2, implies that it suffices to exhibit $|\delta(v)| + 1$ vectors satisfying the conditions of the lemma. There are two cases to consider.

1. If $v \in \bigcup_{k \in \tilde{K}} V_k \backslash W$, assume that $v_{h(v)} = v$. The $|\delta(v)| + 1$ incidence vectors associated with the following trees meet the requirements: $\{\{r, s\}\} \cup star(p, K \backslash \{h(r)\})$ for $p \in V \backslash (V_{h(r)} \cup V_{h(s)} \cup V_{h(v)})$ and $star(p, K)$ where $v_{h(r)} = r, v_{h(s)} = s$ if $p \in \{r, s\}$ and $v_{h(r)} \neq r, v_{h(s)} \neq s$ if $p \in V_{h(r)} \cup V_{h(s)} \backslash \{r, s\}$. Since $|\tilde{K}| \geq 2$, there exists q with $h(q) \in \tilde{K}, h(q) \neq h(v)$ and the last vector can be $\{\{r, s\}, \{r, v\}\} \cup star(q, K \backslash \{h(r), h(s)\})$.

2. If $v \in (V_{h(r)} \cup V_{h(s)}) \backslash \{r, s\}$, assume that $h(v) = h(r)$. The $|\delta(v)| + 1$ incidence vectors associated with the following trees meet the requirements: $\{\{v, p\}, \{v, v_{h(s)}\}\} \cup star(v_{h(s)}, K \backslash \{h(p), h(r)\})$ where $v_{h(s)} \neq s$ for $p \in V \backslash \bigcup_{k \in \tilde{K} \cup \{h(r), h(s)\}} V_k$ and $\{\{v, p\}\} \cup star(v_{h(s)}, K \backslash \{h(r)\})$ where $v_{h(s)} \neq s$ for $p \in V$ with $h(p) \in \tilde{K}$ and $star(v, K)$ where $v_{h(s)} = p$ for $p \in V_{h(s)}$. Let $\tilde{k} \in \tilde{K}$, the last vector can be associated with $star(v, K \backslash \{h(s)\}) \cup \{\{v_{\tilde{k}}, v_{h(s)}\}\}$ where $v_{h(s)} = s$.

□

4 Branch-and-cut algorithm, tabu search heuristic, and separation procedures

The branch-and-cut algorithm we have developed for the GMSTP is based on the undirected cluster subpacking formulation described in Section 2. Since there are exponentially many GSEC (3), these are initially relaxed and generated whenever they are found to be violated. In addition, other valid inequalities may also be introduced. While carrying out preliminary experiments, we have found that violated Hammock inequalities (8) are seldom identified and therefore we have not used them in our final tests. Identifying Odd Cycle and Odd Clique-Matching for a general value of m inequalities is rather difficult and designing even a heuristic separation procedure for this case does not seem to pay in computational terms. Note however, that Odd Cycle and Odd Clique-Matching inequalities coincide when $m = 3$. For this particular case we have devised a heuristic separation procedure and the violated inequalities turned out to be rather efficient. The Odd Hole inequalities are also useful and can be separated quite effectively. Whenever a feasible solution is reached, a heuristic procedure attempts to improve it in the hope of obtaining a better incumbent. At a fractional solution, a rounding procedure is called with the aim of obtaining a better feasible solution.

4.1 Summary of the branch-and-cut algorithm

Step 1 *(Initialization).* The linear program

$$\begin{aligned} & \text{Minimize} && \sum_{\{i,j\}\in E} c_{ij}x_{ij} \\ \text{subject to} \quad & && y(V_k) = 1, \\ & && x(E) = |K| - 1, \\ & && 0 \le x_e \le 1 && e \in E, \\ & && 0 \le y_v \le 1 && v \in V, \end{aligned}$$

is inserted in a list L. The best known objective value z_{best} is obtained by means of a tabu search heuristic (described in Section 4.2).

Step 2 *(Termination check and subproblem selection).* If L is empty, stop. Otherwise extract one subproblem from L according to a best-first rule. Different ways of selecting a subproblem from the list were tested.

Step 3 *(Subproblem solution).* Solve the subproblem using an LP-solver and let z be its optimal value. If $z \ge z_{best}$, go to step 2 . Otherwise, if this solution is feasible, call the local improvement procedure. If this solution is fractional, call the rounding procedure (see Subsection 4.3).

Step 4 *(Separation procedure: GSEC (6) and (7)).* Search for violations of GSEC (6) and (7). If violations are identified, introduce the corresponding constraints into the subproblem and go to Step 3.

Step 5 *(Separation procedure: other GSEC).* Search for violations of GSEC (3). If violations are identified, introduce the corresponding constraints into the subproblem and go to Step 3.

Step 6 *(Separation procedure: other valid inequalities).* Search for violations of Odd Cycle inequalities (9) for $m = 3$, and Odd Hole inequalities (11). If violations are identified, introduce the corresponding constraints into the subproblem and go to Step 3.

Step 7 *(Branching).* Create two new subproblems by branching on a constraint $y(V_k) = 1$. Add these subproblems to the list L and go to Step 2.

In the following subsections, we explicit some of the steps of this algorithm. We denote by $G^*(V^*, E^*)$ the support graph associated with a given fractional solution (x^*, y^*) where $V^* = \{i \in V : y_i^* > 0\}$, and $E^* = \{\{i, j\} : x_{ij}^* > 0\}$.

4.2 Tabu search heuristic

The tabu search heuristic used in Step 1 to determine an initial solution is now described.

Several construction heuristics can be designed for the GMSTP by adapting one of the available exact polynomial algorithms for the *Minimum Spanning Tree Problem* (MSTP). Three such algorithms were developed by respectively Kruskal, Prim and Sollin (see Ahuja, Magnanti and Orlin [2]). All were implemented, tested and the adaptation of Kruskal's algorithm turned out to be the best for the GMSTP. In the case of the MSTP, edges are first sorted in non-decreasing order of their cost. Kruskal's algorithm then selects edges in that order provided that no cycle or conflict is created.

Given a feasible GMSTP solution created by this procedure, a tabu search process can be put in place to explore the solution space. Starting from an initial GMSTP solution s_1, this process moves, at iteration t, from a solution s_t to the best non-tabu solution s_{t+1} in the neighbourhood $N(s_t)$ of s_t yielding the best solution value $f(s_{t+1})$. Since $f(s_{t+1})$ is not necessarily better than $f(s_t)$, a anti-cycling mechanism must be implemented. More specifically, all solutions possessing some attribute of s_t are declared forbidden, or *tabu*, for θ iterations. In our implementation, $N(s_t)$ is obtained by deleting a vertex v contained in s_t and replacing it by another vertex v' in $V_{h(v)} \backslash \{v\}$; a new GMSTP solution is then computed by using $(V(s_t) \cup V_{h(v)}) \backslash \{v\}$, where $V(s_t)$ is the vertex set of s_t. Several criteria were used for the choice of v but none was better than a random choice. We used $\theta = 8$ and ran the algorithm for $1000|V|$ iterations. Extensive tests showed that the initial solution could not in general be improved whenever $|V|/|K| < 3$.

4.3 Local improvement and rounding procedure

Whenever a new best known feasible solution is identified in Step 3 of the branch-and-cut algorithm, an attempt to improve it is made. This consists of performing a local search, as described in the tabu search heuristic, without allowing neighbour solutions that do not improve upon the current solution. Whenever a fractional solution is identified in Step

3 of the algorithm, an attempt is also made to improve it. For this a GMSTP solution is computed on the support graph of the current solution, using the modified Kruskal algorithm described in Subsection 4.2, where the cost of an edge $\{i,j\}$ in E^* is equal to $1-x^*_{ij}$.

4.4 Branching

Branching on a constraint is performed as in Williams [21]. Two descendants of the current subproblem are created as follows. Assume that there exists at least one vertex $i \in V$ such that y^*_i is not integer. Let V_k be a cluster with at least one vertex i such that $0 < y^*_i < \frac{1}{2}$. This cluster is divided into two parts $V^1_k$and V^2_k such that $y^*(V^1_k)$ and $y^*(V^2_k)$ are close to $\frac{1}{2}$. The two son subproblems are then generated from the current subproblem in adding $y(V^1_k) = 0$ for one and $y(V^1_k) = 1$ for the other. Preliminary tests have shown that this branching outperforms the standard rule consisting of branching on the x variable closest to $\frac{1}{2}$ and with maximum objective coefficient.

We now proceed to the description of the separation procedures applied in Steps 4, 5 and 6.

4.5 Separation of Generalized Subtour Elimination constraints

For each $v \in V$, we check whether $x(\delta(v)) \geq y_v$ is violated. For each $v \in V$ we search for the cluster for which $x(E(\{v\} : V_k)) \leq y_v$ is the most violated. Adding for each $v \in V$ all violated constraints of the form $x(E(\{v\} : V_k)) \leq y_v$ (not only the most violated) was found to be less efficient in preliminary tests. This inequalities are declared *static* i.e., they always remain in the linear program.

Other GSECs are separated as follows. For every cluster $k \in K$ we find a subset S_k containing V_k for which the constraint $x(E(S)) \leq y(S) - 1, V_k \subset S \subset V, k \in K$ is violated. This is done by solving a maximum flow problem on an auxiliary undirected graph with capacities depending on the current solution (x^*, y^*) and on k (see Magnanti and Wolsey [15] for this construction for the MSTP). The max flow algorithm used in our implementation is taken from Goldberg and Tarjan [11]. For $k \in K$ the auxiliary graph $\tilde{G}(\tilde{V}, \tilde{E})$ is defined as follows. Let $\tilde{V} = V^* \cup \{s,t\}$ and $\tilde{E} = E^* \cup \{\{s,v\} : v \in V^*\} \cup \{\{v,t\} : v \in V^*\}$. The capacity for every edge $e \in E^*$ is $x^*_e/2$, the capacity of an edge $\{s,v\} \in \tilde{E}$ is equal to $x^*(\delta(v))/2$ and the capacity of an edge $\{v,t\} \in \tilde{E}$ is equal to y^*_v if $v \in V^* \backslash V_k$ and to zero if $v \in V^* \cap V_k$. This construction is depicted in Figure 5. It is important to note that $\tilde{G}$ is connected. Indeed, the current solution (x^*, y^*) is supposed, at this point, to satisfy constraints (6) and (7). Using constraint $x(E) = |K| - 1$ and $x(E) = x(E(S)) + x(E(V \backslash S)) + x(\delta(S))$, the inequality $x(E(S)) \leq y(S \backslash V_k)$ is equivalent to $|K| - 1 \leq x(E(V \backslash S)) + x(\delta(S)) + y(S \backslash V_k)$. Let S be a subset containing s and not containing t. We show that the capacity of an (s,t)-cut of $\tilde{G}$ is equal to $x^*(E(V \backslash S)) + x^*(\delta(S)) + y^*(S \backslash V_k)$. The subset S satisfies $\{s\} \subset S \subset \tilde{V} \cup \{s,t\}$. The edges $\{s,v\}$ with $v \in V^* \backslash S$ give a capacity equal to $\sum_{v \in V^* \backslash S} x^*(\delta(v))/2 = x^*(\delta(S))/2 + x^*(E(V \backslash S))$.

The edges $\{v_1, v_2\}$ with $v_1 \in V^* \cap S, v_2 \in V^* \backslash S$ give a capacity equal to $x^*(\delta(S))/2$. The edges $\{v, t\}$, $v \in S$ give a capacity equal to $y^*(S \backslash V_k)$. Let S^* be the subset of V^* corresponding to the minimum cut. If the value of this cut is strictly less than $|K|-1$, then $x^*(E(S^*)) > y^*(S^* \backslash V_k)$. The GSEC of type (3) corresponding to the subset $S_k = S^* \cup V_k$ is generated. Indeed, this constraint is violated, $x^*(E(S^*)) > y^*(S^* \backslash V_k)$ implying that $x^*(E(S^* \cup V_k)) > y^*(S^* \cup V_k) - 1$ since $x^*(E(S^* \cup V_k)) \geq x^*(E(S^*))$ and $y^*(S^* \cup V_k) - 1 = y^*(S^* \backslash V_k)$. If the value of the minimum cut is greater than or equal to $|K| - 1$, then no GSEC of type (3) is violated. Indeed such a GSEC corresponds to an (s, t)-cut in $\tilde{G}$ induced by a subset S containing V_k. Clearly, such an (s, t)-cut has a capacity at least as large as the minimum cut. At each iteration, at most $|K|$ constraints are added.

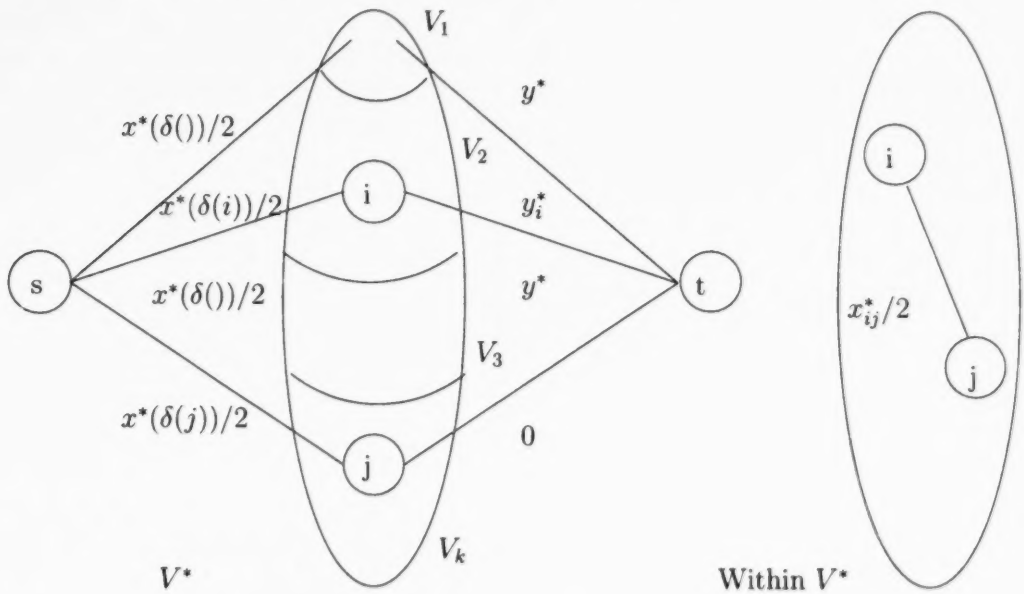

Figure 5: Undirected auxiliary graph for GSEC separation

4.6 Separation of Hammock inequalities

The Hammock inequalities can be separated in polynomial time. Whenever $x_{rs}^* > 0$, we determine the set $\tilde{K} \subset K \backslash \{h(r), h(s)\}$ for which the Hammock constraint corresponding to r, s is the most violated. Let $z_q = 1$ if $q \in \tilde{K}$ and 0 otherwise. The separation problem associated with the current solution (x^*, y^*) is then to find a solution such that

$$\Big(\sum_{q \in K \backslash \{h(r), h(s)\}} z_q - 1\Big) x_{rs}^* + y_r^* + y_s^* + \sum_{q \in K \backslash \{h(r), h(s)\}} z_q x^*(E(V_q : V_{h(r)} \cup V_{h(s)} \backslash \{r, s\})) +$$
$$x^*(E(V_{h(r)} \backslash \{r\} : V_{h(s)} \backslash \{s\})) - \Big(\sum_{q \in K \backslash \{h(r), h(s)\}} z_q + 1\Big)$$

is maximum. If the optimal value of

$$\max_{z_q \in \{0,1\}, q \in K \setminus \{h(r),h(s)\}} \sum_{q \in K \setminus \{h(r),h(s)\}} (x^*_{rs} + x^*(E(V_q : V_{h(r)} \cup V_{h(s)} \setminus \{r,s\})) - 1) z_q \qquad (26)$$

is strictly greater than

$$1 - y^*_r - y^*_s - x^*(E(V_{h(r)} \setminus \{r\} : V_{h(s)} \setminus \{s\})) + x^*_{rs}, \qquad (27)$$

then a maximally violated inequality corresponding to $\{r, s\}$ has been identified. Problem (26) can be solved trivially.

4.7 Separation of Odd Cycle inequalities and Odd Clique-Matching inequalities for $m = 3$

As mentioned above, we have only separated Odd Cycle inequalities and odd clique-matching inequalities for $m = 3$, in which case, they coincide. The heuristic we have devised for this separation can be described as follows.

Step 1. Determine $k, l, m \in K, k \neq \ell \neq m$ with $x^*(E(V_k \cup V_\ell \cup V_m))$ maximum. We assume in the following that $k = 1, \ell = 2$ and $m = 3$.

Step 2. For all $v \in V_1 \cup V_2 \cup V_3$ in the support graph of the current solution, compute

$$val(v) = x^*(E(\{v\} : V_{(h(v) \bmod 3)+1})) - x^*(E(\{v\} : V_{((h(v)+1) \bmod 3)+1})).$$

This value is the difference between the weight of the edges from v to the next cluster in the 3-cycle and the weight of the edges from v to the previous cluster in the 3-cycle. Let L be the list of the vertices in $V_1 \cup V_2 \cup V_3$ already set to V^1 or V^2, $L := \emptyset$. We denote by V^*_{123} the subset of the vertices of $V_1 \cup V_2 \cup V_3$ in the support graph.

Step 3. While$(V^*_{123} \cap L \neq V^*_{123})$ do

1. $v^* = \arg \max_{v \in V^* \setminus L} |val(v)|$,
2. if $val(v^*) > 0$, set v^* to $V^1_{h(v)}$, else set v^* to $V^2_{h(v)}$, $L := L \cup \{v^*\}$,
3. update $val(v)$. For all $v \in V^*_{123} \setminus L$ do
   (a) if $v^* \in V^1_k$, $k \in \{1, 2, 3\}$, for all $v \in V_{((k+1) \bmod 3)+1}$ *(the previous cluster in the 3-cycle)* do $val(v) := val(v) - x^*_{v^*v}$;
   (b) if $v^* \in V^2_k$, $k \in \{1, 2, 3\}$, for all $v \in V_{(k \bmod 3)+1}$ *(the next cluster in the 3-cycle)* do $val(v) := val(v) + x^*_{v^*v}$.

Step 4 All the vertices in $(V_1 \cup V_2 \cup V_3) \setminus V^*$ are arbitrarily set to one or another part of the partition of their cluster. For instance, if $v \in V_1$, v is set to V^1_1, if $v \in V_3$, v is set to V^1_3; if $v \in V_2$, v is set to V^2_2.

4.8 Separation of Odd Hole inequalities

To identify violated Odd Hole inequalities, we simply apply the procedure for determining maximum odd weight cycle described in Grötschel, Lovász, Schrijver [13].

5 Computational results

The branch-and-cut algorithm just described was coded in C++ and uses the ABACUS 2.3 library (Thienel [20], [1]) with CPLEX 7.0 as LP-solver (see ILOG [3]). All tests were run on a Generic sun4u sparc SUNW, Ultra-5.10 workstation (360 MHz,128 Mo RAM) running under SunOS 5.8 operating system. The algorithm was tested on randomly generated instances and on other instances, including those of TSPLIB (Reinelt [19]). We first describe the generation of the test problems, followed by the experiments themselves.

5.1 Instance generation

Two types of randomly generated instances were used. *Euclidean* instances were obtained by randomly generating $|V|$ points in the $[0, 100]^2$, according to a continuous uniform distribution. The cost of edge $\{i, j\}$ was defined as the Euclidean distance between i and j. *Non-Euclidean* instances were obtained by randomly generating a cost matrix according to a continuous distribution on $[0, 100]$. In both cases, vertices were randomly assigned to clusters. Tests were performed on graphs having different edge densities γ. To ensure feasibility, we first randomly generated a spanning tree on the clusters. Additional edges were then randomly generated to reach the required density. Random instances with $24 \leq |V| \leq 200$ and $8 \leq |K| \leq 40$ were tested.

We also performed tests on all TSPLIB instances with $48 \leq |V| \leq 226$. We used the same clustering procedure as in Fischetti, Salazar and Toth [9]. The number of clusters is equal to $|K| = \lceil |V|/5 \rceil$. The $|K|$ *centers* are determined by selecting $|K|$ vertices with a large geographical dispersion. The first center is chosen randomly, the second center is the farthest vertex from the first center, the third center is the farthest vertex from the subset including the two previous centers and so on. The clusters are finally obtained by assigning each vertex to its nearest center. In addition, we worked on five additional instances. The first two (15spain47 and 27europ47) correspond to 47 cities and 15 clusters (regions) of Spain, and to 47 cities and 27 clusters (countries) of Europe. The last three are the geographical problems gr96 (Africa), gr137 (America) and gr202 (Europe) of Grötschel and Holland [12]. In these five instances, clusters correspond to regions (in 15spain47) or to countries otherwise.

5.2 Summary of computational results

Our main computational results are summarized in Tables 1 to 4. The table headings are as follows.

Name:	problem name for geographical and TSPLIB instances (the digits correspond respectively to the value of $\|K\|$ and $\|V\|$);
$\|K\|$:	number of clusters;
$\|V\|$:	number of vertices;
$\|E\|$:	number of edges;
γ:	graph density;
Succ:	number of instances solved to optimality within 7200 seconds out of 5 instances;
Opt:	optimal objective value for geographical and TSPLIB instances.
Pair:	number of generated constraints (6) and (7);
GSEC:	number of generated GSEC(3) with $\|S\| \geq 3$;
Oddcy:	number of generated Odd Cycle inequalities (9) with $m = 3$;
Oddho:	number of generated Odd Hole inequalities (11);
Nodes:	total number of nodes generated in the branch-and-cut tree;
LB%:	best lower bound reached before the branching as a percentage of the optimal value;
UB%:	best upper bound reached before the branching as a percentage of the optimal value;
LB/UB %:	lower bound as a percentage of upper bound before branching;
Myung LB/UB %:	lower bound as a percentage of upper bound before branching obtained in Myung et al. [16];
Seconds:	CPU time in seconds.

Tables 1 to 3 present average statistics over 5 instances except for LB% and UB% which are computed over the number of successful instances. The Myung LB/UB% column is provided in order to compare our results with those of Myung et al. [16]. For the instances that are not solved within the time limit, the column LB/UB% represents the lower bound as a percentage of the best upper bound found so far. In the Myung LB/UB% column, *** means that this type of instance was not reported by Myung et al.[16]. Since we did not have access to the instances nor to the code of these authors, the reported values are those reported by Myung et al. [16] for instances having the same size as ours and generated in the same way. Thus the comparison is only indicative since the instances are not identical.

Results presented in Table 1 indicate that our algorithm can solve random Euclidean instances involving up to 160 vertices within reasonable computing times. Table 2 shows that random non-Euclidean instances are much easier to solve than Euclidean instances; all instances (up to 200 vertices) tested for this class are solved to optimality within less than two hours. Table 3 indicates that for both classes, computational difficulty decreases with the graph density.

Tables 1, 2 and 3 also show that the GSEC (including (6) and (7)) are the most generated inequalities, followed by Odd Cycle and Odd Hole inequalities. The latter constraints tend to be used more often in larger size instances ($|V| \geq 100$). Preliminary tests have

shown that Hammock inequalities are very seldom generated so that they were left out in our final implementation.

The LB/UB% column for random instances indicates that the gap between the lower bound and the upper bound obtained before branching is better, by 10% to 20% than the value reported by Myung et al. [16] except for one case with $|V| = 120$ and $|K| = 40$ in Table 1. The computing times spent by our algorithm are larger than those of Myung et al. [16]. This can be justified by the high quality of our lower bound. Moreover we solve the root node LP to optimality while Myung et al. only solve a relaxation.

All but four of the geographical and TSPLIB instances were solved within two hours at the root node of the search tree using only GSECs (including (6) and (7)). The four unsolved instances are 40d198, 41gr202, 45ts225 and 46pr226.

Table 1: Average computational results for complete random Euclidean instances

$\|K\|$	$\|V\|$	$\|E\|$	Succ	Pair	GSEC	Oddcy	Oddho	Nodes	LB%	UB%	LB/UB %	Myung LB/UB %	Seconds
8	24	252	5	61.6	44.6	0.0	0.2	1.4	99.8	100.0	99.8	89.6	0.2
	32	448	5	95.4	88.2	0.0	0.2	1.4	99.5	100.0	99.5	93.8	1.4
	48	1008	5	153.6	107.8	0.0	0.0	1.0	100.0	100.0	100.0	91.5	2.6
	80	2800	5	277.6	191.8	0.0	0.0	1.0	100.0	100.0	100.0	90.5	14.4
10	30	405	5	89.6	115.2	0.4	1.2	1.4	99.7	100.0	99.7	84.9	1.2
	40	720	5	132.6	173.0	0.4	0.2	1.4	99.5	100.0	99.5	90.8	2.8
	60	1620	5	228.6	290.4	0.0	0.0	1.4	99.8	100.0	99.8	89.7	10.2
	100	4500	5	424.4	477.2	0.0	0.2	1.0	100.0	100.0	100.0	89.4	61.0
12	36	594	5	117.6	176.6	0.0	0.2	1.0	100.0	100.0	100.0	90.1	2.4
	48	1056	5	178.8	289.2	0.0	0.0	1.0	100.0	100.0	100.0	87.0	5.2
	72	2376	5	309.0	519.8	0.4	0.2	1.4	99.6	100.4	99.2	85.5	26.0
15	45	945	5	155.8	280.0	0.0	0.0	1.4	99.9	100.6	99.3	88.5	5.6
	60	1680	5	232.2	411.0	0.0	0.0	1.0	100.0	100.0	100.0	86.5	11.8
	180	15120	3	998.0	4818.6	0.8	3.2	3.8	* * *	* * *	97.4	83.4	4775.2
	120	6720	5	565.2	1602.6	0.2	0.4	2.2	99.8	100.8	99.0	86.2	299.8
	150	10500	5	791.4	2968.4	1.2	1.8	7.4	99.5	101.7	98.0	86.0	1204.6
18	54	1377	5	217.8	708.4	0.0	4.8	6.2	99.2	100.6	98.7	89.9	16.8
20	120	6840	5	640.0	3790.6	0.0	3.0	7.8	99.1	102.2	97.1	83.8	600.4
	160	12160	5	983.6	9139.0	0.2	1.8	6.6	98.9	104.7	94.7	80.0	3925.6
	200	19000	1	1108.8	7163.6	0.4	0.0	1.8	* * *	* * *	95.4	79.0	6547.8
25	150	10800	5	862.6	7112.0	0.0	2.6	9.0	99.2	105.0	94.9	82.2	1607.6
	200	19200	0	1156.0	10569.6	0.0	0.6	1.4	* * *	* * *	84.9	74.6	7200.0
30	120	6960	5	1037.4	17665.4	3.2	18.8	32.2	97.8	109.2	90.1	81.2	1456.6
	150	10875	5	1058.6	18982.8	0.8	14.8	17.4	97.5	112.7	86.7	74.8	2864.8
	180	15660	4	1079.0	16480.2	0.2	3.0	6.6	* * *	* * *	98.2	81.0	5069.0
40	120	7020	5	652.0	10351.8	0.0	8.0	17.0	98.9	117.0	84.6	88.2	697.8
	160	12480	3	1336.2	42142.0	1.0	25.2	31.4	* * *	* * *	94.2	79.6	6098.4
	200	19500	0	1211.4	23454.6	0.2	2.2	5.4	* * *	* * *	79.7	77.7	7200.0

Table 2: Average computational results for complete instances with random costs

$\|K\|$	$\|V\|$	$\|E\|$	Pair	GSEC	Oddcy	Oddho	Nodes	LB%	UB%	LB/ UB %	Myung LB/UB %	Seconds
8	24	252	34.0	1.2	0.0	0.0	1.0	100.0	100.0	100.0	100.0	0.0
	32	448	43.2	4.2	0.0	0.0	1.0	100.0	100.0	100.0	96.1	0.2
	48	1008	94.8	47.4	0.0	0.6	1.0	100.0	100.0	100.0	90.4	1.4
	80	2800	99.0	15.4	0.0	0.0	1.0	100.0	100.0	100.0	94.9	4.2
10	30	405	54.0	8.2	0.0	0.0	1.0	100.0	100.0	100.0	87.3	1.0
	40	720	56.0	8.2	0.0	0.0	1.0	100.0	100.0	100.0	96.0	1.0
	60	1620	101.6	23.2	0.2	0.2	1.0	100.0	100.0	100.0	83.3	3.2
	100	4500	158.0	20.8	0.0	0.0	1.0	100.0	100.0	100.0	88.0	8.8
12	36	594	60.4	13.0	0.0	0.0	1.0	100.0	100.0	100.0	84.8	1.8
	48	1056	100.8	101.6	0.4	0.6	1.8	100.0	100.8	99.2	92.8	3.2
	72	2376	172.6	84.8	0.0	0.0	1.0	100.0	100.0	100.0	88.4	6.8
15	45	945	78.8	31.8	0.0	0.0	1.0	100.0	100.0	100.0	85.7	3.6
	60	1680	125.0	69.0	0.0	0.0	1.8	99.2	102.4	97.1	85.5	6.2
	120	6720	242.6	50.6	0.0	0.0	1.0	100.0	100.0	100.0	89.0	21.4
	150	10500	266.8	213.0	0.0	0.8	1.4	100.0	101.3	98.8	86.7	42.4
	180	15120	275.2	48.0	0.0	0.0	1.0	100.0	100.0	100.0	82.7	46.8
18	54	1377	104.6	108.2	0.0	0.2	2.2	100.0	100.5	99.5	92.3	7.6
20	120	6840	294.4	376.4	0.0	0.2	1.8	100.0	104.5	96.3	81.4	39.8
	160	12160	422.4	1212.8	0.0	0.2	1.4	100.0	103.0	97.4	76.4	122.2
	200	19000	431.6	1225.4	0.6	3.4	4.6	100.0	106.3	94.6	77.5	191.4
25	150	10800	412.8	2518.6	0.2	2.8	6.2	98.5	112.7	88.3	80.5	178.8
	200	19200	444.0	361.0	0.4	1.6	1.4	100.0	102.5	97.8	81.0	140.6
30	120	6960	304.0	597.6	0.0	0.4	1.4	100.0	104.6	96.3	78.3	61.0
	150	10875	371.2	1876.2	0.6	2.8	3.0	100.0	106.7	95.0	83.9	152.6
	180	15660	367.8	146.4	1.0	0.6	1.8	100.0	104.1	96.6	74.3	114.6
40	120	7020	315.8	547.0	0.0	0.0	1.0	100.0	100.0	100.0	84.1	92.6
	160	12480	457.8	4540.6	0.0	0.6	3.8	100.0	106.8	94.2	74.9	288.6
	200	19500	450.0	2152.2	0.0	0.6	2.2	100.0	109.7	92.2	79.6	302.2

Table 3: Average computational results for Euclidean and random cost instances with different densities

			Euclidean costs						Random costs		
			$\gamma = 1$		$\gamma = 1/2$		$\gamma = 3/10$		$\gamma = 1$	$\gamma = 1/2$	$\gamma = 3/10$
$\|K\|$	$\|V\|$	$\|E\|$	Succ	Seconds	Succ	Seconds	Succ	Seconds	Seconds	Seconds	Seconds
8	24	252	5	0.2	5	0.0	5	0.0	0.0	0.0	0.0
	32	448	5	1.4	5	0.0	5	0.2	0.2	0.0	0.0
	48	1008	5	2.6	5	1.2	5	0.6	1.4	0.6	0.2
	80	2800	5	14.4	5	5.0	5	3.4	4.2	2.2	3.2
10	30	405	5	1.2	5	0.6	5	0.0	1.0	0.0	0.0
	40	720	5	2.8	5	1.0	5	0.6	1.0	1.0	0.6
	60	1620	5	10.2	5	3.2	5	1.2	3.2	1.6	1.2
	100	4500	5	61.0	5	14.8	5	15.4	8.8	4.8	3.8
12	36	594	5	2.4	5	1.4	5	1.0	1.8	1.0	1.0
	48	1056	5	5.2	5	3.2	5	1.2	3.2	1.2	1.0
	72	2376	5	26.0	5	10.0	5	4.4	6.8	3.8	2.2
15	45	945	5	5.6	5	3.4	5	2.2	3.6	2.0	1.4
	60	1680	5	11.8	5	7.2	5	4.2	6.2	3.2	2.2
	180	15120	3	4775.2	5	121.6	5	40.8	21.4	17.8	10.8
	120	6720	5	299.8	5	307.2	5	163.0	42.4	46.8	15.4
	150	10500	5	1204.6	5	973.8	5	304.4	46.8	45.2	51.8
18	54	1377	5	16.8	5	7.6	5	4.4	7.6	4.8	2.8
20	120	6840	5	600.4	5	129.2	5	83.4	39.8	21.0	13.2
	160	12160	5	3925.6	5	951.2	5	1600.4	122.2	40.4	132.0
	200	19000	1	6547.8	3	4632.2	5	1545.4	191.4	227.6	64.2
25	150	10800	5	1607.6	5	1756.8	5	782.0	178.8	75.0	32.2
	200	19200	0	7200.0	3	5806.0	4	2642.2	140.6	539.2	99.2
30	120	6960	5	1456.6	5	300.2	5	154.4	61.0	37.6	33.2
	150	10875	5	2864.8	5	1445.8	5	234.4	152.6	48.6	134.0
	180	15660	4	5069.0	5	1966.6	5	1585.2	114.6	207.8	160.8
40	120	7020	5	697.8	5	241.0	5	195.4	92.6	63.2	33.8
	160	12480	3	6098.4	5	3088.8	5	917.4	288.6	171.4	284.2
	200	19500	0	7200.0	1	7051.0	4	5433.0	302.2	1558.8	133.6

Table 4: Computational results for geographical and TSPLIB instances

Name	$\|E\|$	Opt	Pair	GSEC	Seconds
15spain47	985	2393	136	225	5
27europ47	1042	13085	127	620	3
50gr96	4463	306	291	5950	68
35gr137	8251	209	299	1748	110
34gr202	19018	135	971	16456	4558
10att48	1010	10923	144	194	4
10gr48	1017	1282	125	156	3
10hk48	995	4119	162	205	4
11eil51	1158	132	159	154	5
12brazil58	1464	9206	227	416	14
14st70	2248	233	251	596	20
16eil76	2660	186	294	976	47
16pr76	2661	46514	264	960	37
20gr96	4292	221	361	2262	99
20rat99	4609	402	371	1500	83
20kroa100	4727	7982	370	1423	65
20krob100	4716	8111	338	1400	73
20kroc100	4714	8041	356	1740	87
20krod100	4716	7643	393	2820	183
20kroe100	4702	8164	379	1220	66
20rd100	4703	2779	337	1180	55
21eil101	4776	204	354	1200	76
21lin105	5130	6728	383	2751	109
22pr107	5441	20398	423	4958	244
24gr120	6820	2255	426	1944	114
25pr124	7315	30174	466	6790	753
26bier127	7191	58150	532	5434	908
28pr136	8879	34104	564	4340	406
28gr137	8892	329	557	10976	1518
29pr144	9952	40055	527	11609	861
30kroa150	10809	9815	557	4290	426
30krob150	10807	10048	571	8305	849
31pr152	11080	39109	617	13268	1541
32u159	12031	18723	547	6720	592
39rat195	18478	751	743	12246	2120
40kroa200	19409	11634	749	20600	2607
40krob200	19430	11244	767	28240	5254

6 Conclusion

We have proposed a compact formulation and an exact branch-and-cut algorithm for the Generalized Minimum Spanning Tree Problem, a network design problem with applications in the field of telecommunications. The model uses an undirected cluster subpacking formulation to which several classes of valid inequalities are incorporated. We have shown several of these inequalities to be facet defining and we have proposed heuristic separation procedures within the framework of the branch-and-cut algorithm. A simple yet effective tabu search heuristic was also proposed for the problem. Computational experiments were conducted on randomly generated graphs with Euclidean and non-Euclidean costs. We also used some geographical instances and several problems derived from TSPLIB. Tests have shown that Euclidean instances involving up to 160 vertices and random cost instances containing up to 200 vertices can be solved optimally, the latter class being much easier to solve. All but four of the geographical and TSPLIB instances were also solved. Our tests show that the proposed algorithm outperforms the only other known exact algorithm for the GMSTP developed by Myung et al. [16]. On comparable instances our lower bound to upper bound ratio at the root of the search tree is much higher than that of Myung et al., and we can also solve much larger instances.

Aknowledgements. This work was partially supported by the Fonds pour la Formation à la Recherche dans l'Industrie et dans l'Agriculture, by the Région Bruxelles-Capitale, by the Direction générale des Relations Extérieures de la Région wallonne de Belgique (99.13424) and by the Canadian Natural Sciences and Engineering Research Council under grant OGP0039682. The research of the second author was done in part during a visit at the Institut de Mathématiques of the Ecole Polytechnique Fédérale de Lausanne. This support is gratefully acknowledged.

References

[1] Universität zu Köln, Universität Heidelberg. OREAS (Optimization Research and Software). *ABACUS, A Branch-And-CUt System. Version 2.3. User's Guide and Reference Manual*, Köln (1999).

[2] R.K. Ahuja, T.L. Magnanti, J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice-Hall, Englewood Cliffs, NJ (1993).

[3] ILOG CPLEX 7.0. (2000).

[4] M. Dror, M. Haouari, J. Chaouachi. *Generalized Spanning Trees.* European Journal of Operational Research **120** (2000) 583–592.

[5] U. Faigle, W. Kern, P.C. Pop, G. Still, *The Generalized Minimum Spanning Tree Problem*, Working Paper, Departement of Operations Research and Mathematical Programming, University of Twente, **http://www.math.utwente.nl/dos/ormp/preprints.htm** (2000).

[6] C. Feremans, M. Labbé, G. Laporte. *On Generalized Minimum Spanning Trees.* European Journal of Operational Research **134** (2001) 457–458.

[7] C. Feremans, M. Labbé, G. Laporte. *A Comparative Analysis of Several Formulations for the Generalized Minimum Spanning Tree Problem.* Networks **39** (2002) 29–34.

[8] M. Fischetti, J.J. Salazar, P. Toth. *The Symmetric Generalized Traveling Salesman Polytope.* Networks **26** (1995) 113–123.

[9] M. Fischetti, J.J. Salazar, P. Toth. *A Branch-and-Cut Algorithm for the Symmetric Generalized Traveling Salesman Problem.* Operations Research **45** (1997) 378–394.

[10] M.X. Goemans. *The Steiner Tree Polytope and Related Polyhedra.* Mathematical Programming **63** (1994) 157–182.

[11] A.V. Goldberg, R.E. Tarjan. *A New Approach to the Maximum Flow Problem.* Journal of Association for Computing Machinery **35** (1988) 921–940.

[12] M. Grötschel, O. Holland. *Solution of Large-Scale Symmetric Traveling Salesman Problems.* Mathematical Programming **51** (1991) 141–202.

[13] M. Grötschel, L. Lovász, A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag, Berlin (1988).

[14] E. Ihler, G. Reich, P. Widmayer. *Class Steiner Trees and VLSI-design.* Discrete Applied Mathematics **90** (1999) 173–194.

[15] T.L. Magnanti, L.A. Wolsey, *Optimal Trees*, Handbooks in Operations Research and Management Science, M.O. Ball, T.L. Magnanti, C.L. Monma, G.L. Nemhauser (Editors), Elsevier Science, Amsterdam, Vol. 7, Chap. 9 (1995) 503–615.

[16] Y.S. Myung, C.H. Lee, D.W. Tcha. *On the Generalized Minimum Spanning Tree Problem.* Networks **26** (1995) 231–241.

[17] M. Padberg. *On the Facial Structure of Set Packing Polyhedra.* Mathematical Programming **5** (1973) 199–215.

[18] P.C. Pop, W. Kern, G.J. Still. *An Approximation Algorithm for the Generalized Minimum Spanning Tree Problem with Bounded Cluster Size.* Working Paper, Departement of Operations Research and Mathematical Programming, University of Twente, **http://www.math.utwente.nl/dos/ormp/preprints.htm** (2001).

[19] G. Reinelt. *TSPLIB - A Traveling Salesman Problem Library.* ORSA Journal on Computing **3** (1991) 376–384.

[20] S. Thienel. *ABACUS, A Branch-And-CUt System.* Ph.D. dissertation, Universität zu Köln (1995).

[21] H.P. Williams. *Model Solving in Mathematical Programming.* Wiley, Chichester (1992).